



Hansel minutes

Hanselminutes is a weekly audio talk show with noted web developer and technologist Scott Hanselman and hosted by Carl Franklin. Scott discusses utilities and tools, gives practical how-to advice, and discusses ASP.NET or Windows issues and workarounds.

### Text transcript of show # 56

March 22, 2007

#### The Old New Thing - Interview with Raymond Chen

Scott sits down with the original raconteur of Windows, Raymond Chen to talk about all things Win32 and where Microsoft Bob is currently.

(Transcription services provided by [PWOP Productions](http://www.pwop.com))



#### Our Sponsors

 **telerik**

*deliver more than expected*

<http://www.telerik.com>

 **nsoftware**

<http://www.nsoftware.com/>

**NET**  **DEVELOPER'S JOURNAL**

<http://dotnet.con-sys.com>





## The Old New Thing - Interview with Raymond Chen March 22, 2007

**Lawrence Ryan:** From [hanselminutes.com](http://hanselminutes.com), it's Hanselminutes, a weekly discussion with Web developer and technologist, Scott Hanselman, hosted by Carl Franklin. This is Lawrence Ryan, announcing show #56, recorded Thursday, March 15<sup>th</sup>, 2007. Support for Hanselminutes is provided by [telerik r.a.d. controls](http://telerik.com), the most comprehensive suite of components for Windows Forms and ASP.NET Web applications, online at [www.telerik.com](http://www.telerik.com). Support is also provided by [/n software, Red Carpet Subscriptions](http://nsoftware.com), the most comprehensive solution for adding connectivity to your .NET and ASP.NET Web applications, with components for every major Internet Protocol, online at [www.nsoftware.com](http://www.nsoftware.com); and by [.NET Developer's Journal - The World's leading .NET developer magazine](http://www.sys-con.com), online at [www.sys-con.com](http://www.sys-con.com). In this episode, Scott talks with Windows raconteur Raymond Chen.

**Scott Hanselman:** Hi, this is Scott Hanselman and this is another episode of Hanselminutes and I'm sitting here in the actual office of the actual Raymond Chen, the author of, '*The Old New Thing*'. He has just signed my copy of the book and I can die now. Raymond, how are you sir?

**Raymond Chen:** I am doing great; and I hope that that copy fetches a nice bounty on eBay.

**Scott Hanselman:** I have already put it on eBay, actually. And it's doubly listed on Craig's list. I intend to sell this for as much money as possible; I might get two or three bucks over retail.

**Raymond Chen:** I am impressed. That's more than what I'd have expected.

**Scott Hanselman:** I appreciate that. So I'm a huge fan of your blog and I actually -- I don't know if you put two and two together, but if you open up your book and you look on the first page...

**Raymond Chen:** Oh, heavens, I should read that.

**Scott Hanselman:** That's the craze for *The Old New Thing* on the first page. Raymond Chen is the original raconteur of Windows. I went to the thesaurus at Merriam Webster, found out what 'raconteur' meant, and said -- I thought it was actually inappropriate, but it actually did mean exactly what it is. You are the keeper of the stories.

**Raymond Chen:** A story teller.

**Scott Hanselman:** In fact you are. And I love the book, and when they said, "Hey, put a quote in

the front", I said I got to be a part of that. I really appreciate your taking the time to talk to us here. I know that you've talked to Carl and the guys over at .NET Rocks! for an hour or so, and you had Carl in stitches for the better part of an hour. But I want to get down to the brass tacks here and find out some of the fundamentals about, what you work on? You're doing Win32 in the Shell, right?

**Raymond Chen:** Yeah, so I've been for -- I don't even remember when I came over to the User-Interface Group in the Shell team, so this is the -- we're the people who deal with Explorer and common dialogs and that sort of thing. It is kind of hard to describe where the boundaries are to non-technical people. Well, you have a technical audience so, I think...

**Scott Hanselman:** I do have a technical audience.

**Raymond Chen:** It is mostly Shell32 and Com Dialog, and Com Control and...

**Scott Hanselman:** Ooh, Com Control.

**Raymond Chen:** ...things that hang out in that world.

**Scott Hanselman:** I remember the great 3-D wars of the late -- mid to late 90's when there were 15 -16 different versions of -- was it control...

**Raymond Chen:** Control 3D, Control 3D 2.0, Control 3D 2.32.

**Scott Hanselman:** What's the deal man? And hand in sub-versions within those; why was that such a problem?

**Raymond Chen:** Because if 3D is good, more 3D must be better.

**Scott Hanselman:** Absolutely. I had a lot of trouble with that. That was DLL Hell; that was the beginning of it, wasn't it?

**Raymond Chen:** I don't know; I was asleep for that part.

**Scott Hanselman:** Is it really difficult to do versioning like that? You're just exporting a function table, how hard is it to keep those functions looking the same? Why do we get these big -- I guess they weren't called GPS back in the day, they were Unhandled Application Exceptions - AUEA - UAE's, Unrecoverable Application Error...



## The Old New Thing - Interview with Raymond Chen March 22, 2007

**Raymond Chen:** That's right. Wow! That brings back horrible memories. And you've to make sure not to confuse it with the United Arab Emirates.

**Scott Hanselman:** In fact yes, the people in Dubai do not appreciate that in any way.

**Raymond Chen:** That people don't like UAE's for some reason. It's like, "We are just minding our own business."

**Scott Hanselman:** Why was versioning so hard? Why were there so many copies of that DLL?

**Raymond Chen:** There were so many copies of that DLL. I actually don't know, because I wasn't around for that, but I assume that the reason there were so many copies of that DLL was, everybody thought it was really awesome - or everybody wanted to have their own version of it, because, well, mine is cooler than yours. But I realize that I wrote it, it must necessarily be cooler than yours.

**Scott Hanselman:** Yes, 'invented here; syndrome.

**Raymond Chen:** Yes, exactly.

**Scott Hanselman:** Specifically invented here. Now, I realize that I would definitely -- I will probably ask you a few questions. I know that you're known for volleying back questions that are not -- I wasn't here, sorry, I was sick that day.

**Raymond Chen:** I am very good at rejecting questions.

**Scott Hanselman:** And I appreciate that. So most of this interview will be rejected questions. But I'm more interested in...

**Raymond Chen:** You can listen for them on the deleted scenes.

**Scott Hanselman:** In fact yes, on the special features. I am interested in versioning because you're living in a non-managed world and I haven't lived there in a while. I fuked the last -- last time I fuked was at least ten years ago and I appreciate the occasional...

**Raymond Chen:** Scars are healed.

**Scott Hanselman:** In fact, I appreciate the occasional -- the reference pointer and that's great that you guys do that, but is versioning really hard in an unmanaged world, and then you've got Methods inside of Shell32 and the different things that you maintain that probably

you wish you could yank out, but you just can't for compatibility, right?

**Raymond Chen:** Oh, there's stuff that we want to yank out all the time, but we publish the Function, we document it, now we're stuck with it.

**Scott Hanselman:** Now, from a Static Dependency Analysis perspective, you can keep those functions looking the same, things will line up, they take the right parameters, but is it really difficult to make sure that the semantics of those functions, including bugs, stays the same?

**Raymond Chen:** Yeah, bug-for-bug compatibility is one of the things that you sort of don't appreciate until you have to do it. In fact, looking at the changes between Windows XP and Vista, we see people complaining about things that -- saying that, "Hey, if I pass this invalid string to path is..." and they're like, "path is network path", - it used to say it was, and now it says it isn't. And we say, "That's right because it isn't." "Well, I was sort of hoping it would still be one because now it's slowing down my program."

**Scott Hanselman:** I remember literally the day that I double-clicked on the time-date thing and saw that the -- in the time-date applet, I saw that you had removed the time zones because of that border war or whatever with -- I guess was it Peru or something?

**Raymond Chen:** Right, the time zone one was an Ecuador-Peru Border Wars, I recall.

**Scott Hanselman:** And I was sad; and I said, a feature has been removed and I have seen it happen - I noticed. I am sure that a lot of people didn't notice. I know that you said on Carl's talk, that they remove scraps and then you remove scraps personally...

**Raymond Chen:** I remove scraps.

**Scott Hanselman:** I noticed damn it. And I -- no I didn't really notice. No, I'm kidding.

**Raymond Chen:** I don't think anybody has even noticed yet.

**Scott Hanselman:** I actually noticed when I make a scrap, and I go, "What the hell is that?"

**Raymond Chen:** What was that? How do I get rid of this thing?

**Scott Hanselman:** And I got that immediately off my desktop. I didn't even open one up once; put to the MZ header to see if it was evil.



## The Old New Thing - Interview with Raymond Chen March 22, 2007

**Raymond Chen:** Yeah, I suspect that most people who created scraps did it completely by mistake, and all the support calls were questions of the form "How do I get rid of this thing?"

**Scott Hanselman:** Fantastic. The semantics - you said that the semantics are very difficult to maintain. I understand that there is some kind of a --and I may be misspeaking here, but there is some kind of a secret, evil, parallel universe inside of a Windows Box, that's like the application compatibility -- we know everything there is to know about Arthur's Teacher Trouble so we will patch up these six APIs.

**Raymond Chen:** There is a not insubstantial database of programs and all the little things that have to be done to keep them happy. And it doesn't even begin to cover all of the applications out there. The AppCompat Team have a rather large library but it's clearly not comprehensive.

**Scott Hanselman:** So how are they doing that? They are literally saying, Arthur's Teacher Trouble doesn't work anymore; and it's because they are passing nonsense to get path info or something.

**Raymond Chen:** Those guys are amazing; they have to have a pretty good knowledge of large chunks of the system. When a program stops working, they are usually the ones, who do the initial debugging to figure out what's going on, and then once they narrow it down, they can hand it off to a particular component team if they determine that there is a shell functionality change that's causing trouble, then they can send it over our way, but they have a pretty tough job at having to figure out even what area the problem even lies in.

**Scott Hanselman:** So specific to the Shell, when I go Properties, Compatibility, and I see all of that stuff, it seems like I'm doing that a lot now in Vista, I'm saying, "No no, XP Service Pack 2.0." What exactly is happening there? You are patching, Get Version?

**Raymond Chen:** That is, when you pick one of those larger categories; like if you run the -- I'm trying to remember what it's called, the Application Compatibility Toolkit.

**Scott Hanselman:** Yeah where is that it's like app compatwiz.exe?

**Raymond Chen:** It's -- search your favorite search engine, and hopefully it will turn up. The Application Compatibility Toolkit lets you apply individual fixes, and there are hundreds upon hundreds of them to a specific application, but if

you use the Compatibility Page then you're sort of -- it's sort of like a combo meal, where we sort of, have the Windows XP combo meal, of if you pick this, we'll apply these 25 fixes that tend to cover most of the problems that application have had. They can range from just reporting that it's Windows XP, even though it really is Windows Vista, to checking applications to see if they try to access a file by an old file name, we turned it into the new file name before sending it through.

**Scott Hanselman:** So, it's lying.

**Raymond Chen:** Its various degrees of sleight of hand.

**Scott Hanselman:** So, back in the day, when men were men and Program Manager reigned supreme -- I think when Windows 95 - you came in a Windows 95 kind of universe, right? Was it Chicago or was it Memphis?

**Raymond Chen:** It was called - what was it called -- it was called Cougar back then.

**Scott Hanselman:** Oh my!

**Raymond Chen:** Yeah it was -- that was actually a precursor project to what eventually became Windows 95.

**Scott Hanselman:** They were -- the icons were prettier. They were less -- back in the day before Windows XP, before Windows 2000 there were fewer device contexts, fewer brushes. There was a minimal amount of stuff that you could paint. I remember, every once in a while, when my computer got really stressed out, icons would started turning black.

**Raymond Chen:** Ah yes, the black icons.

**Scott Hanselman:** Why?

**Raymond Chen:** Because you ran out of GDI resource and somebody tries...

**Scott Hanselman:** But you also had a black brush though - I mean, someone was able to why not paint them white, why black?

**Raymond Chen:** Well, but the way icons are drawn, you have your mask and then you have -- if you recall, a bitmap -- an icon back in the day consisted of two bitmaps; a monochrome mask bitmap, and a color bitmap and the algorithm was to and with the mask and then XOR with the color bitmap. And so, the mask contained black pixels, everywhere you wanted color to show through. So, if you ran into a problem with the color part, all you got was the mask part which -- all it did



was, set pixels to black. The job of the mask was to set pixels to black and the job of the color part was to start coloring them in again; and so, if you ran out of resources in the color in part, all you got was stuff getting set to black.

**Scott Hanselman:** Now, the icons now...

**Raymond Chen:** It's through the magic of Boolean Algebra, that old "And" thing; it's really easy to make things black.

**Scott Hanselman:** Yeah. Didn't get that at community callers; I'm really looking at this -- Boolean Algebra, let's all Google for that, find out what that is

**Raymond Chen:** I think in the new mryhod, it's just called 'True False Algebra'.

**Scott Hanselman:** Oh! So, like an 'if' statement.

**Raymond Chen:** 'If' is too judgmental; we just like to call it 'whenever'.

**Scott Hanselman:** Raymond has a standup career on the side in his mind. A lot of people pick on Explorer; they pick on it a lot, and I'm sure you must get that.

**Raymond Chen:** Like I pick on it a lot.

**Scott Hanselman:** Well, some people call it the Explorer, sometimes it crashes. One of the things I get frustrated with is when some one -- I always like to anthropomorphize...

**Raymond Chen:** Back in the old days, we also called it File Damager.

**Scott Hanselman:** Oh, that's brutal dude. We're at Microsoft campus and we're talking like this. It's bad.

**Raymond Chen:** We actually, over lunch or over beer or over beer at lunch, often, as many people do, spend our time, trying to come up with derogatory terms for products around the world; and it turns out there are a lot of Microsoft products that we're familiar with so we spend a lot of time coming up with derogatory names for them. Part of this is, just to desensitize ourselves so that when we hear it out in the wild we don't feel so bad.

**Scott Hanselman:** That's cool. So tell me why Explorer seems to lock up particularly when it's enumerating drives; I open up the little plus, it goes off, and then suddenly, oop, sorry, your CD -- I don't know what's wrong with your CD, your

kid put a sandwich in it, and I can't show the next drive.

**Raymond Chen:** The sandwich in the CD - that was me actually, I would say. And part of it is that, the Shell is heavily based on COM, and specifically Apartment Model COM, and one of the advantages of Apartment Model COM is that everything happens on a single thread in COM -- well, make sure that your Objects -- and in fact, you have to observe the rule that your Objects are on the thread they are created on, which makes it very easy to implement your Objects. The downside of this is that it's kind of, stuck to a thread. And we're kind of stuck with this design because that's what we have had since Windows 95; there are thousands upon thousands of Shell extensions out there that are relying on Apartment threading - and pull incredibly evil tricks that break down if we ever put them on some other thread. So we're kind of stuck with putting a lot of things on the UI thread. We've been working to move things off but -- and in Windows Vista - in fact in Windows XP as I recall, at least Folder Enumeration was done in the background and I believe tree -- I can't remember whether Tree Expansion was done in the background or not, but I definitely remember that back in the old days when the address bar was just a drive's combo box, that when you click that, it took time to spin up all the drives to get the icons and the volume information because that is clearly pending upon an action; you click the button, the dropdown has to drop down, and if it drops down, it's got to have stuff in it. So, you are kind of stuck on that one but...

**Scott Hanselman:** Does it have to have stuff out? Can it say, I'm working on it? Like for example, I know that Mark Versonowitch did a big thing on his blog recently where, when he is a domain user and he is not on the domain, Explorer hangs for 11 seconds while someone tries to figure out what his real name is, so they can go and say, see users, domain, my name, Mark Versonowitch - his name/My Documents, and when he is off the domain, he is not connected, he has no network connection. It hangs, realizes he is not on the network, 11 seconds later it goes, see users Mark R, the actual folder name. So it's...

**Raymond Chen:** Part of the problem with that, it sort of, goes back to the behavior changes and the function design upfront is that, somebody calls, they give me the name for this thing and you have to give him an answer. You can't say -- you could, but as it was originally -- as the function was originally designed, the guy asks you, what's the name for this folder and you give him an answer, and if you decide to add a new



## The Old New Thing - Interview with Raymond Chen March 22, 2007

answer called, "I don't know" it would take too long" or...

**Scott Hanselman:** Try a little later...

**Raymond Chen:** This might take a while; you should do this on a background thread. Then all the people who don't know about this behavior will just start getting empty strings back.

**Scott Hanselman:** So, if the assumption is that I've got a limited number of milliseconds before I got to get this drop down back to the user, and if UI responsiveness is the number one priority, forgetting for a moment about the inconvenience, or just procedure calls in general, in the sense that they have to have a well known return type. If someone could institute a -- if you can't get this back to me in 300 milliseconds then just stop, bail and I'll never talk to you again. And I'll make something like the default answer.

**Raymond Chen:** But of course that requires you to be able to predict the future; will this call to get username take more than 300 milliseconds?

**Scott Hanselman:** So this is the Heisenberg Uncertainty Principle of dropdown lists?

**Raymond Chen:** No, it's more like, life would be a lot easier if we had time machines.

**Scott Hanselman:** It would.

**Raymond Chen:** And there are plenty of things I would change.

**Scott Hanselman:** Like what?

**Raymond Chen:** I don't know, because on the other hand I might accidentally destroy myself, it's kind of a...

**Scott Hanselman:** You would probably kill your own grandfather.

**Raymond Chen:** Yeah, accidentally, I would sneeze and create this horrific epidemic or something.

**Scott Hanselman:** So going back -- so thinking about time machines for a second, everyone thought -- and by everyone, within the context of this room, that would be me, thought that all the icons in Vista would be vectors -- and I think I thought that -- either someone said it, or they said they will be scalable - and I had a flashback to prodigy and I said, "Oh, awesome - vectors, those are great, I hear those are doing good things." So, I just assume that there would be

some slider bar somewhere and I would just gleefully slide it back and forth with my giant icons going bigger and smaller and bigger and smaller. And in fact in Vista, the icon formats got to be quite a package.

**Raymond Chen:** The icon format has actually changed not too much and that's -- and I think we may have changed it too much even then. The icon format has been stable since the beginning of Windows...

**Scott Hanselman:** So I could put a 1024 x 1024 icon in an icon format?

**Raymond Chen:** Up until Windows Vista, when we added PNG icons, that was the first incompatible change to the icon format. Through Windows XP, you could take those icons and whack them on a Windows 1.0 machine and you'll be able to come up with icons...

**Scott Hanselman:** So, the icon format, it's a bag of icons. You can have as many icons as you want in ICO.

**Raymond Chen:** It's an icon directory and then there is a header that describes how many images there are, and what their sizes are and their color depth; and all of that has remained the same up through Windows XP, then Windows 2003. Windows Vista, we took a chance on adding a new icon format and in fact our first attempt was to fit it in with the -- to sort of follow, how the original icon format sort of was meant to be with a header and then the bits and have a different...

**Scott Hanselman:** Could you put different bits in there?

**Raymond Chen:** Well, we were actually going to set that because there in that info header there is a Compression Type field. So we put png in the Compression Type field, and it turns out that created a lot of compatibility problems.

**Scott Hanselman:** Probably the third party vendors, bit map libraries looked at that info and they said, oh...

**Raymond Chen:** There were some very important -- there was a very popular icon-editing tools, that when faced with icons that had Compression Type of png, just exploded. And it turns out that that's not something we sort of enjoy.

**Scott Hanselman:** So you just changed the format?



**The Old New Thing - Interview with Raymond Chen**  
March 22, 2007

**Raymond Chen:** So we had to go back in and redo the format a little, so that way, the file format was -- I guess the problem was that the way we originally put ping support in, it looked too much like an old-fashioned icon; and that tricked these programs into thinking that, "Oh, I know it; I know what I'm doing." And so, in fact the solution to compatibility was ironically to make things slightly less compatible and get these programs to sort of panic and back out before they got into too much trouble.

(00:20:15)

**Scott Hanselman:** So you designed the format such that they would...

**Raymond Chen:** We actually designed the format so it didn't follow the intended expansion design of the icon format. It's somewhat sad.

**Scott Hanselman:** But also in such a way that -- well, it's clever though in the sense of -- it's like the GIF format -- and I don't say JIF - I don't believe that, that's right.

**Raymond Chen:** I don't think many people even say it much anymore.

**Scott Hanselman:** Well, I just said it.

**Raymond Chen:** Okay.

**Scott Hanselman:** And I'll say it again. So the GIF format -- I remember back in the day, they added notations at the end, like GIF89a, stuck a bunch of stuff, even putting notes on the end of your GIF. And it wasn't part of the original format or something, so basically they figured out that if we put it here, nobody will look here and they'll hit the null and they'll go, "Oh that's cool". I don't think it's so fun to make those kind of -- we made this backwards compatible in that no one blows up when we read it.

**Raymond Chen:** Because we put it in a place that most people consider to be junk anyway.

**Scott Hanselman:** Exactly. There is a lot of hunting for junk that goes on when you need to expand the file format or -- especially one of these older file formats that everybody knows in quotation marks, where you would think that oh, well, we can just use this bit over here and that indicates a new format over there but now, because everybody knows this file format.

**Scott Hanselman:** It's hard to make a binary -- to expand a binary format, it's kind of like, you don't know about this, but because you don't

know about anything that's not in the Shell. You refuse to answer all questions -- I was going to mention Office; of the Office format, it took structured storage and then basically in the Office format, as you make a doc x file, you go to Word, type "Hello World" save as, fu.docx. It's a zip file. You can rename it .zip, open it up, there's a whole directory structure in there.

**Raymond Chen:** And there is stuff in there.

**Scott Hanselman:** What a clever way to make an expandable format...

**Raymond Chen:** That's really smart.

**Scott Hanselman:** It is. But there's all of the overhead of the zipping and the zip header and all that kind of stuff from the point of view of an icon.

**Raymond Chen:** And then you are also stuck with -- you can't add any new compression formats at this point.

**Scott Hanselman:** Exactly, so at some point the binary bits have to be -- you're stuck with it. You're not going to go and make the ARJ or the RAR

**Raymond Chen:** Any of those other fantastic acronyms that mean nothing to me.

**Scott Hanselman:** Really that mean anything to you? You are a DOS TSR guy, then went DeskView back in the day. You just like appeared in -- at Microsoft in 1990...

**Raymond Chen:** I can tell a story about DeskView.

**Scott Hanselman:** Let's hear it.

**Raymond Chen:** DeskView is the SysRq key. That's -- you have this keyboard, there's a button called SysRq - what is that for? That's for DeskView. The purpose of SysRq, as it was originally designed, as I understand it, was to be sort of, a way to get the attention of the controlling operating system. So, it was sort of like the old-fashioned idea of what we now call the secure attention sequence, the thing that only the operating system listens to and...

**Scott Hanselman:** Like control-alt-delete.

**Raymond Chen:** Yeah, so -- and as I understand it, DeskView is the only program that actually used SysRq to mean okay, I want to call up the DeskView menu thing and nobody else paid SysRq any attention at all. But there it is on your



**The Old New Thing - Interview with Raymond Chen**  
**March 22, 2007**

keyboard, sitting around, waiting for somebody to press it, and then of course you press it and nothing happens, because no program cares about it either.

**Scott Hanselman:** Can you do something with it? I mean I've never seen a...

**Raymond Chen:** You can draw a smiley face on it.

**Scott Hanselman:** I've never seen a hotkey program say that would be great; they overload the hell out of the Windows key, but I've got a brand new Lenovo laptop here with a little blue function key and SysRq is just there waiting for me. I thought it was like -- so there's Print Screen and Alt Print Screen. I thought it had something to do with that.

**Raymond Chen:** No, no, Print Screen used to be on the asterisk key; it was -- back in the ancient days -- this is why you always had to have a printer hooked up to your computer and ready because if you can go -- you can go search on the Internet, find pictures of those really, really old keyboards, and the right-hand Shift key was not a huge key like it is today; it was the size of a normal key, and next to it to the right, was a key that had an asterisk when un-Shifted and was Print Screen when Shifted. So, if you were a sloppy typist, your right pinky would reach for the Shift key, it would slip too far and it would hit the Shift key and the asterisk - which meant Print Screen.

**Scott Hanselman:** That's good design.

**Raymond Chen:** Yeah, so you would be sitting there, minding your own business, typing away and then your machine would lock up because it's waiting for your printer...

**Scott Hanselman:** nnnnnnnnn

**Raymond Chen:** No it didn't make any noise...

**Scott Hanselman:** That was that printer sound. Remember the old dot matrix printers, they would nnnnnnnnnnnnnnn,

**Raymond Chen:** Oh, the dot -- I just remember inkinkk sound, I don't remember the head moving around, not doing anything.

**Scott Hanselman:** That's good design. You read "*The Design of Everyday Things*", I'm sure you probably have that around here somewhere - "*The Teapot Book*."

**Raymond Chen:** Yes, it's a fantastic book.

**Scott Hanselman:** I love that kind of stuff. There's such legendary design to applications that just really push the limit - like Microsoft Bob.

**Raymond Chen:** Oh, Bob! Yeah, Bob was a product ahead of its time, it was clearly -- it was sort of too revolutionary for its own good, it's...

**Scott Hanselman:** I read it; I sincerely tried for a week.

**Raymond Chen:** Its genius was underappreciated and has now been supplanted by YouTube. But believe it or not, Bob is not dead. Everybody, if you have your Windows XP CD, there is a copy of Bob on that CD. What is a copy of Bob doing on my Microsoft Windows XP CD? So, after consulting with all the translators and all the other people who would need to have space on the CD, we noticed that, well, there was some empty space on the CD. So, we paid for the CD; we may as well put stuff on it. And as sort of one of the lame attempts at slowing down people who like to make illegal copies, we were going to -- we put some dummy files or dummy data on the CD, and Setup would check that the dummy data was still there to make it -- so that way, if you were to make a copy of the CD, you still had to make a copy including all the dummy data. And when people are transferring files on 56k modems, an extra 30 meg of dummy data sort of slows you down - not that that slows people down much anymore. And the dummy data is Microsoft Bob, the person who had to come up with the dummy data was looking around for about -- he was like, I need -- I don't remember whether it was 30 meg or something, it was -- but he said, I need a corpus of dummy data; where am I going to find some dummy data? And he could have called CryptGenRandom and generated 10 meg of CryptGenRandom data but there is no fun in that. So instead, he went to the internal release servers, went through the archives, and found a copy of 'Microsoft Bob' and took all the Microsoft Bob floppies which comes out to -- I don't know, whatever size he needed -- but of course, Microsoft Bob floppies aren't very random, so he encrypted them by typing -- smashing his hand on the keyboard and generating keys that he will never be able to reproduce again - and out came an encrypted copy of Microsoft Bob, which has the nice property that the contents are closer to random - it's still not cryptographically random, but it was messed up enough, and that's what went out into the unused spaces on the CD as balance data. So, if you go and find your copy of Windows XP that you bought in the store and you know where to look -- and I myself don't know where, because I just heard this story -- and if



you can somehow mash your hand on the keyboard in exactly the right way, out will come a copy of Microsoft Bob.

**Scott Hanselman:** It's interesting.

**Raymond Chen:** So Microsoft Bob -- one might say that Windows XP has been the most effective Microsoft Bob delivery system.

**Scott Hanselman:** Bob is everywhere.

**Raymond Chen:** Bob is everywhere.

**Scott Hanselman:** It's stories like that that make the book so great -- your book, *The old new thing*, that some people are calling the 'new old' thing.

**Raymond Chen:** I'm sort of learning to get used to it. I've...

**Scott Hanselman:** Why is it called 'The old new thing?'

**Raymond Chen:** The title of the blog was in fact determined in about 35 seconds. Brad Abrams said, "Raymond, you should start doing this blog thing" and I said, wow, you really think so, I'm not sure. And he said yeah, just do it. And so, I started filling out the paperwork to set up the blog, and one of the questions is, so what's its title? And I said, "I didn't realize, it needed a title." And I just made up the phrase, 'The old new thing', on the principle that I was going to be writing about old stuff, but stuff that at one time was new. And so 'the old new thing' is talking about things that back in the day were the new thing. So that's sort of where the name comes from.

**Scott Hanselman:** Well, it's huge; I know, I was just sitting here and we were looking at your stats and you're becoming more and more popular. People enjoy reading it. The book has new content and you've revised some of the stories. It's very well organized. I guess, even Joel Spolsky, Joel on Software, the pundits are noticing you, you are becoming a power broker in the blogosphere.

**Raymond Chen:** Yeah, I can always tell whenever Joel makes a reference to one of my stories because all of a sudden they start showing up in all these other data mining stuff. And like recently, he retold the story that I shared, and it's in the book, it's on the blog, about 'blowing the dust out of the connectors', a way to get people to double-check things -- as sort of the 'catch all' description for getting people to check things that they would normally resist

checking on the assumption that it would make them look stupid; you would say, "Is it plugged in?" - and of course they're going to say, "Well, of course it's plugged in, I'm not that stupid." - and they're not going to check. You have a pretty good feeling that it's not plugged in, so you tell them, "Well, sometimes the connector gets kind of dusty, could you get down there, unplug it, blow into the connector to get the dust out and then plug it back in. So maybe that will clear it up." And they'll go down there, they'll notice that it's not plugged in, or if it was plugged into the wrong plug, they'll unplug it, they blow into the connector, then they plug it back in. Now they're paying attention; now they're going to plug it into the correct port. They come back, and now it works. And they think you're great, and everybody is happy and nobody loses face. I remember there was another story about a different type of support call; somebody had called in to report a geopolitical error in the maps used in flight simulator. And if you've been following what I've been writing, that maps -- and we hit upon it earlier, maps are extremely sensitive data. In general, if you can get away without using a map, that's often the safest bet; but if you're a flight simulator and the object at the game is to fly around the world, you kind of have to have maps. And somebody had called in to report that there was some sort of border problem on the map, and the report came in, it went through the normal bug report analysis thing, and the person got the response saying that, thank you for reporting the problem, we've taken a look at it and we believe that everything is just fine, that the map is correct. This person was sort of unsatisfied with this answer, and instead of going through the normal Escalation procedures, sent mail to Bill Gates. Bill reads his email -- or at least, enough of it gets to him, that when this issue came up to Bill Gates, he went back to the Flight Sim team and said, "I want to know what's going on here." And the Flight Sim team were like, "Okay, we thought we had taken care of this, but apparently this person decided to just take that case straight to the Supreme court." They went back and looked at it again and called the guy back and tried to get more information about why he thinks the map is wrong, which map they're using, because we're using this map, and where do you think the border should be? And at the end of the day it turned out that this person was basing his map information upon a map of the world on the shower curtain that he had in his bathroom. Said shower curtain was a couple of years old and hadn't been kept up-to-date with the latest changes in borders in Eastern Europe.

**Scott Hanselman:** That's awesome. And Bill was involved.



**Raymond Chen:** So, this would normally not have been a problem except you bring in Bill and everybody goes on high alert.

**Scott Hanselman:** And he still doesn't answer my emails.

**Raymond Chen:** Yeah, well, you need to stop opening them with like, "Hey loser."

**Scott Hanselman:** Oh, that's my problem.

**Raymond Chen:** Yeah.

**Scott Hanselman:** Well, listen Raymond, I really appreciate you taking the time out of your day, I know that you are busy-busy. You've got all sorts of stack dumps and stuff up on your screen here. I know you're busy out there saving lives.

**Raymond Chen:** Yes, I put -- stack comes up just to look impressive.

**Scott Hanselman:** Just look impressive. Yeah, there's highlights and stuff so they go, "Yeah that's the offset into whatever yeah." Then it's like any read-assembler with just do with my eyes closed. But listen, the book is fantastic and I don't just say that because the author is sitting right here. I really enjoyed it and it's a -- I probably shouldn't say this, but it's a great bathroom book because it's got little short stories.

**Raymond Chen:** That was actually kind of the goal; it was not meant to be a book, you sit down and plough through all at one go. It's meant to be a book that you read as time permits.

**Scott Hanselman:** And that's exactly what it's allowing me to do. I'm really enjoying the stories. I think it's great and I really appreciate your time. Thanks a lot.

**Raymond Chen:** Thank You.